

 Stagnant Standards Track: ERC

ERC-1388: Attestation Issuers Management List

Authors Weiwu Zhang <a@colourful.land>, James Sangalli <j.l.sangalli@gmail.com>

Created 2018-09-08

Discussion Link <https://github.com/ethereum/EIPs/issues/1388>

Table of Contents

- Introduction
- Purpose
- Draft implementation
- Related ERC's

Introduction

In smart contracts, we will need methods to handle cryptographic attestations to a users identifier or abilities. Let's say we have a real estate agent, KiwiRealtors, that provides an "expression of interest" function though a smart contract and requires the users to provide an attestation that they are a resident of New Zealand or Australia, as a legal requirement. This has actually happened in the New Zealand property market and it is the perfect example of a need to handle such attestations.

However, it is not practical for a smart contract to explicitly trust an attestation issuer. There are multiple issuers who can provide an attestation to a person's residency - a local Justice of the Peace, the land title office, local police, passport authority etc. We envision a model where the effort to manage the list of qualified issuers is practically outsourced to a list.

Anyone can publish a list of issuers. Only the most trusted and carefully maintained lists gets popular use.

Purpose

This ERC provides a smart contract interface for anyone to manage a list of attestation issuers. A smart contract would explicitly trust a list, and therefore all attestations issued by the issuers on the list.

Draft implementation

```

/* The purpose of this contract is to manage the list of attestation
 * issuer contracts and their capacity to fulfill requirements
 */
contract ManagedListERC
{
    /* a manager is the steward of a list. Only he/she/it can change the
     * list by removing/adding attestation issuers to the list.

     * An issuer in the list is represented by their contract
     * addresses, not by the attestation signing keys managed by such a
     * contract.
     */
    struct List
    {
        string name;
        string description; // short description of what the list entails
        string capacity; // serves as a filter for the attestation signing key
        /* if a smart contract specifies a list, only attestation issued
         * by issuers on that list is accepted. Furthermore, if that
         * list has a non-empty capacity, only attestations signed by a
         * signing key with that capacity is accepted. */

        address[] issuerContracts; // all these addresses are contracts, no sign
        uint expiry;
    }

    // find which list the sender is managing, then add an issuer to it
    function addIssuer(address issuerContractAddress) public;

    //return false if the list identified by the sender doesn't have this issuer i
    function removeIssuer(address issuerContractAddress, List listToRemoveIssuerFr

    /* called by services, e.g. Kiwi Properties or James Squire */
    /* Loop through all issuer's contract and execute validateKey() on
     * every one of them in the hope of getting a hit, return the
     * contract address of the first hit. Note that there is an attack
     * method for one issuer to claim to own the key of another which
     * is mitigated by later design. */
    //Loop through the issuers array, calling validate on the signingKeyOfAttesto
    function getIssuerCorrespondingToAttestationKey(bytes32 list_id, address signi

    /* for simplicity we use sender's address as the list ID,
     * accepting these consequences: a) if one user wish to maintain
     * several lists with different capacity, he or she must use a
     * different sender address for each. b) if the user replaced the
     * sender's key, either because he or she suspects the key is
     * compromised or that it is lost and reset through special means,
     * then the list is still identified by the first sender's
     * address.

```

```
*/  
  
function createList(List list) public;  
  
/* replace list manager's key with the new key */  
function replaceListIndex(List list, address manager) public returns(bool);  
  
}
```

[Click here to see an example implementation of this ERC](#)

Related ERC's

#1387 #1386

Citation

Please cite this document as:

Weiwu Zhang <a@colourful.land>, James Sangalli <j.l.sangalli@gmail.com>, "ERC-1388: Attestation Issuers Management List [DRAFT]," *Ethereum Improvement Proposals*, no. 1388, September 2018. [Online serial]. Available: <https://eips.ethereum.org/EIPS/eip-1388>.